

Application Brief: NVRAM vs Flash for program storage

2009-03-23

TINI uses NVRAM for many things

The NVRAM of TStik is shared by the TINI NVRAM file system, TINI heap, and TINI NVRAM code space. The NVRAM is backed up by a lithium battery whose lifetime is a strong function of temperature. It's relatively easy to corrupt TINI's heap (power-cycling it at inopportune moments such as when starting up and reclaiming NVRAM memory). In the case of a corrupted heap, when TINI reboots you will lose the contents of the NVRAM file system, including any NVRAM code files, and your IP configuration (unless it has been saved to flash). It's easy and convenient to FTP your code to TINI's NVRAM during development, but it's not reliable to store your code in NVRAM in the deployment phase of your TINI project. So develop in NVRAM for convenience but deploy in flash for reliability.

Flash is where to store your code when you deploy TINI in the field

TINI permits storing one application. During development this application will be the command shell called "slush", which includes telnet and ftp support. Your own application can replace or include slush, as your needs dictate. If your application replaces slush, it will automatically start up after a reset (at power up or any other reset condition) and begin execution.

Use the TINI Ant task to build for Flash

To convert your TINI application to execute from Flash, use these TiniAnt parameters:

```
flash="yes"  
targetaddress="0x470100"
```

TINIConverter will give you the following warning, which you can safely ignore:
[tini] Warning, address exceeds normal TINI flash memory boundaries! (0x470100)

You now need to use JavaKit (not FTP) to store your code in flash memory. If you will be replacing slush then you should also have stored your IP configuration in flash, since you will no longer be able to use the slush "ipconfig" command to modify your IP settings.

Get some more information at <http://www.tstik.com/apnotes/Flash.html>